

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-078258
(43)Date of publication of application : 23.03.2001

(51)Int.Cl. H04Q 7/38

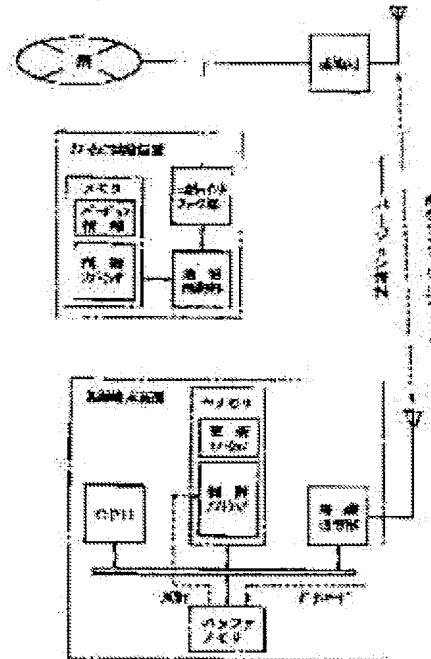
(21)Application number : 11-251065 (71)Applicant : FUJITSU LTD
(22)Date of filing : 06.09.1999 (72)Inventor : IMAMATSU KENICHI

(54) SOFTWARE UPDATE METHOD AND ITS SYSTEM FOR RADIO TERMINAL

(57)Abstract:

PROBLEM TO BE SOLVED: To update a control software efficiently with security through a simple configuration and simple control with respect to a software update method and its system for a radio terminal.

SOLUTION: In the software update method for a radio terminal in a mobile communication system where a base station and radio terminals are interconnected via a radio channel, the radio terminal informs a software supply device connected to a network side about version information of current control software relating to an operation of its own terminal, the software supply device compares informed version information with the newest version information stored and managed by itself to decide the presence of necessity of updating and downloads new control software adapted to revising the informed version to the radio terminal.



(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2001-78258

(P2001-78258A)

(43)公開日 平成13年3月23日(2001.3.23)

(51)Int.Cl.⁷
H 04 Q 7/38

識別記号

F I
H 04 B 7/26
H 04 Q 7/04

テ-マコ-ト⁸(参考)
1 0 9 M 5 K 0 6 7
D

審査請求 未請求 請求項の数10 ○L (全 17 頁)

(21)出願番号 特願平11-251065

(22)出願日 平成11年9月6日(1999.9.6)

(71)出願人 000005223
富士通株式会社
神奈川県川崎市中原区上小田中4丁目1番
1号
(72)発明者 今松 塗一
神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内
(74)代理人 100097087
弁理士 ▲高▼須 宏
F ターム(参考) 5K067 AA33 AA41 BB04 DD17 EE02
EE10 EE22 GG01 GG11 HH22
HH23 KK13 KK15

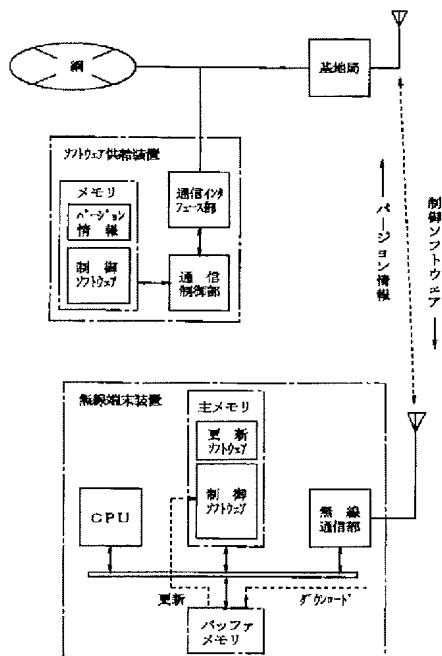
(54)【発明の名称】 無線端末装置のソフトウェア更新方法及びその装置

(57)【要約】

【課題】 無線端末装置のソフトウェア更新方法及びその装置に関し、簡単な構成、制御で制御ソフトウェアの更新を効率良く安全に行えることを課題とする。

【解決手段】 基地局と複数の無線端末装置とが無線回線を介して相互に接続する移動通信システムにおける無線端末装置のソフトウェア更新方法において、無線端末装置は自装置の運用に係る現用の制御ソフトウェアのバージョン情報を網側に接続するソフトウェア供給装置に通知し、ソフトウェア供給装置は通知されたバージョン情報を自己が蓄積管理する最新のバージョン情報と比較して更新の必要性有無を判定すると共に、更新が必要な場合は通知されたバージョンの改訂用に適合する新しい制御ソフトウェアを無線端末装置にダウンロードする。

本発明の原理を説明する図



【特許請求の範囲】

【請求項1】 基地局と複数の無線端末装置とが無線回線を介して相互に接続する移動通信システムにおける前記無線端末装置のソフトウェア更新方法において、無線端末装置は自装置の運用に係る現用の制御ソフトウェアのバージョン情報を網側に接続するソフトウェア供給装置に通知し、これを受けたソフトウェア供給装置は通知されたバージョン情報を自己が蓄積管理する最新のバージョン情報と比較して更新の必要性有無を判定すると共に、更新が必要な場合は通知されたバージョンの改訂用に適合する新しい制御ソフトウェアを選択して無線端末装置にダウンロードすることを特徴とするソフトウェア更新方法。

【請求項2】 無線端末装置は自装置の運用に係る現用の制御ソフトウェアのバージョン情報をソフトウェア供給装置に対して定期的に通知することを特徴とする請求項1に記載のソフトウェア更新方法。

【請求項3】 ソフトウェア供給装置は更新に係る新しい制御ソフトウェアを所定サイズにブロック分割してこれらを断続的にダウンロードすることを特徴とする請求項1に記載のソフトウェア更新方法。

【請求項4】 ディジタル信号を送受信可能な無線通信部と、無線通信制御を含む装置の運用に係る制御ソフトウェア及び該制御ソフトウェアを更新するための更新ソフトウェアを格納する主メモリと、主メモリの各ソフトウェアに従って動作するCPUと、更新用ソフトウェアを一時的に格納するバッファメモリとを備え、

CPUは、現用の制御ソフトウェアの制御下で無線を介してダウンロードされた更新用ソフトウェアをバッファメモリに蓄積して後、更新ソフトウェアの制御下で現用の制御ソフトウェアの対応部分を前記バッファメモリに蓄積した更新用ソフトウェアで更新し、かかる後制御を更新後の制御ソフトウェアに戻すことを特徴とする無線端末装置。

【請求項5】 バッファメモリのサイズは制御ソフトウェアのサイズよりも小さく、かつ更新用ソフトウェアのサイズよりも大きいことを特徴とする請求項4に記載の無線端末装置。

【請求項6】 制御ソフトウェアは各単独で更新可能な複数のモジュールに分割されていると共に、CPUは各モジュールに対するダウンロードと更新の処理を順次繰り返すことにより複数のモジュールを更新することを特徴とする請求項5に記載の無線端末装置。

【請求項7】 更新用ソフトウェアは所定サイズのブロックに分割されていると共に、CPUはブロック単位でダウンロードを要求し、所要サイズ分の更新用ソフトウェアを順次バッファメモリに蓄積することを特徴とする請求項5又は6に記載の無線端末装置。

【請求項8】 CPUはブロックデータの転送順を管理

すると共に、ブロックデータのダウンロード中に所定のイベントが発生した場合はダウンロード処理を中断することを特徴とする請求項7に記載の無線端末装置。

【請求項9】 主メモリの制御ソフトウェアを記憶する部分と、バッファメモリとがフラッシュROMにより構成されていることを特徴とする請求項4に記載の無線端末装置。

【請求項10】 網に接続してデジタル信号を送受信可能な通信インターフェース部と、

網に接続する無線端末装置の運用に係る各バージョンの制御ソフトウェア及びそのバージョン情報を格納するメモリと、

メモリの記憶情報を更新管理すると共に通信インターフェース部を介して任意の無線端末装置と接続可能な通信制御部とを備え、

通信制御部は無線端末装置から通知されたバージョン情報をメモリの最新のバージョン情報を比較して更新の必要性有無を判定すると共に、更新が必要な場合は通知されたバージョンの改訂用に適合する新しい制御ソフトウェアを選択して無線端末装置にダウンロードすることを特徴とするソフトウェア供給装置。

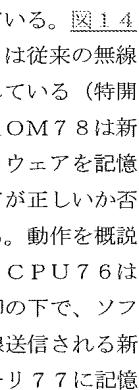
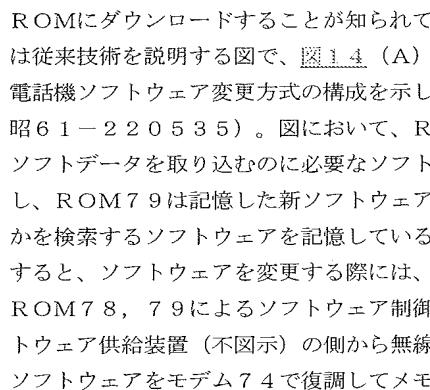
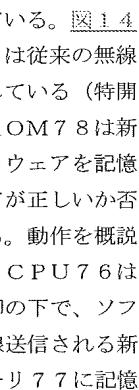
【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は無線端末装置のソフトウェア更新方法及びその装置に関し、更に詳しくは基地局と複数の無線端末装置とが無線回線を介して相互に接続する移動通信システムにおける前記無線端末装置のソフトウェア更新方法及びその装置に関する。

【0002】 今日、移動通信システムにおける通信サービスは急成長を続けており、サービス機能の追加やそれに伴うソフトウェア障害修正等の頻度が高くなっている。これらに要する保守費用の低減等を図るために、携帯電話機等の移動端末装置においても制御ソフトウェアの更新を効率的に行なうことが求められている。

【0003】

【従来の技術】 従来は、制御ソフトウェアを格納しているROM等を交換し、又はプログラマブルROMに外部機器を接続して内容を直接に書き換える方法に代えて、無線回線を介して更改用ソフトウェアをプログラマブルROMにダウンロードすることが知られている。   (A)は従来技術を説明する図で、(B)は従来の無線電話機ソフトウェア変更方式の構成を示している（特開昭61-220535）。図において、ROM78は新ソフトデータを取り込むのに必要なソフトウェアを記憶し、ROM79は記憶した新ソフトウェアが正しいか否かを検索するソフトウェアを記憶している。動作を概説すると、ソフトウェアを変更する際には、CPU76はROM78、79によるソフトウェア制御の下で、ソフトウェア供給装置（不図示）の側から無線送信される新ソフトウェアをモジュール74で復調してメモリ77に記憶

し、これを改めて主記憶データとする。

【0004】しかし、このROM78は無線通信制御（呼制御、送受信制御等）に係る比較的大きなソフトウェアを主メモリ77と重複して記憶しているため、メモリの無駄使いであるばかりか、ROM78のサイズも大きくなり、装置の小型化、低価格化が困難となる。また通信サービスの改訂により、装置の無線通信制御に係るソフトウェアが改訂されると、ROM78を交換しなくてはならない。

【0005】図上部(B)は従来の無線通信装置の構成を示している(特開昭62-38624)。動作を概説すると、EEPROM88に格納されている運用中の制御用プログラムの機能向上を計る場合には、基地局(不図示)から無線送信される更改プログラムを、変復調器86のルートで、かつプロセッサ89及びRAM90によるデータ処理作用を介してEEPROM88に収納する。

【0006】しかし、上記EEPROM88で運用中の現用のプログラムに更改プログラムを上書きする方式であると、ダウンロード(更新)の途中で運用中のプログラムが壊れてしまう場合があり、ソフトウェアの更新を安全に行えない。

【0007】

【発明が解決しようとする課題】上記の如く従来のソフトウェア更新方式は、従前のROM等を交換する方法に代え、無線回線を介して更改用ソフトウェアをダウンロードすることを示すものに過ぎない。しかも、特開昭61-220535では主メモリ77の他に大きなROM78、79を必要としており、また特開昭62-38624ではソフトウェアの更新を安全に行えない。

【0008】本発明は上記従来技術に鑑みなされたもので、その目的とする所は、簡単な構成及び制御で制御ソフトウェアの更新を効率良く安全に行える無線端末装置のソフトウェア更新方法及びその装置を提供することにある。

【0009】

【課題を解決するための手段】上記の課題は例えば図上部の構成により解決される。即ち、本発明(1)のソフトウェア更新方法は、基地局と複数の無線端末装置とが無線回線を介して相互に接続する移動通信システムにおける前記無線端末装置のソフトウェア更新方法において、無線端末装置は自装置の運用に係る現用の制御ソフトウェアのバージョン情報を網側に接続するソフトウェア供給装置に通知し、これを受けたソフトウェア供給装置は通知されたバージョン情報を自己が蓄積管理する最新のバージョン情報と比較して更新の必要性有無を判定すると共に、更新が必要な場合は通知されたバージョンの改訂用に適合する新しい制御ソフトウェアを選択して無線端末装置にダウンロードするものである。

【0010】本発明(1)においては、各バージョンの

制御ソフトウェアをソフトウェア供給装置で集中管理する構成により、制御ソフトウェアの複雑なバージョン管理を網側で安全に一元管理できる。一方、各無線端末装置におけるバージョンの更新は、各無線端末装置の主導で夫々独自に行われるため、ソフトウェア供給装置は無線端末装置毎のバージョン管理を行う必要は無い。また、各無線端末装置におけるバージョンの更新は各無線端末装置からの自発的なアクセス(バージョン情報の通知)を契機とし隨時に行われるため、網側より共通の網設備(基地局等)を介して各無線端末装置に最新のバージョン情報等を報知する必要は無く、既存の網設備に加える変更は殆ど無い。また、ソフトウェア供給装置は呼処理上でも網側に接続する1サーバ装置(端末装置)として扱えるため、この様なソフトウェア更新システムは既存の移動通信システムに容易に導入できる。かくして、本発明(1)によれば、各種バージョンの無線端末装置が混在する様な移動通信システムのバージョン管理を簡単な構成及び制御により能率良く行える。

【0011】好ましくは本発明(2)においては、上記本発明(1)において、無線端末装置は自装置の運用に係る現用の制御ソフトウェアのバージョン情報をソフトウェア供給装置に対して定期的に通知する。

【0012】本発明(2)においては、無線端末装置が定期的に自己のバージョン情報を通知する構成により、無線端末装置の側ではバージョンの改訂(最新バージョン等)に関する事前の知識を要求されない。従って、無線端末装置では別段のバージョン管理が必要無いばかりか、簡単な制御により自装置のバージョンを容易に最新バージョンに維持できる。

【0013】また好ましくは本発明(3)においては、上記本発明(1)において、ソフトウェア供給装置は更新に係る新しい制御ソフトウェアを所定サイズにブロック分割してこれらを断続的にダウンロードする。

【0014】本発明(3)においては、更新用の制御ソフトウェアを所定サイズにブロック分割してダウンロードする構成により、無線端末装置の側では1ブロック分のダウンロードに拘束される時間が短い。また各ブロックを断続的にダウンロードする構成により、その各休止期間を利用して着信を受け付けることも、また発信することも可能となる。従って、無線端末装置の本来の機能は損なわれず、その利便性を維持できる。

【0015】また本発明(4)の無線端末装置は、デジタル信号を送受信可能な無線通信部と、無線通信制御を含む装置の運用に係る制御ソフトウェア及び該制御ソフトウェアを更新するための更新ソフトウェアを格納する主メモリと、主メモリの各ソフトウェアに従って動作するCPUと、更新用ソフトウェアを一時的に格納するバッファメモリとを備え、CPUは、現用の制御ソフトウェアの制御下で無線を介してダウンロードされた更新用ソフトウェアをバッファメモリに蓄積して後、更新ソ

ソフトウェアの制御下で現用の制御ソフトウェアの対応部分を前記バッファメモリに蓄積した更新用ソフトウェアで更新し、かかる後制御を更新後の制御ソフトウェアに戻すものである。

【0016】本発明(4)においては、CPUは現用の制御ソフトウェアの制御下で更新用ソフトウェアを別途バッファメモリにダウンロードする構成により、現用の制御ソフトウェアをフルに活用して、また稼働中の制御ソフトウェアを書き換えることもなく、更新用ソフトウェアを効率良く安全にバッファメモリにダウンロードできる。またダウンロード後は、更新ソフトウェアの制御下で現用の制御ソフトウェア(停止中)の対応部分を前記バッファメモリの内容で更新する構成により、現用の制御ソフトウェアを安全に更新できる。

【0017】好ましくは本発明(5)においては、上記本発明(4)において、バッファメモリのサイズは制御ソフトウェアのサイズよりも小さく、かつ更新用ソフトウェアのサイズよりも大きい。

【0018】本発明(5)においては、バッファメモリのサイズは更新が必要な部分のソフトウェアを記憶できれば十分である。この更新は制御ソフトウェアの部品化されたモジュール毎に行っても良いし、最小1バイト又は1ビットを書き換える様なパッチでも良い。但し、バッファメモリが制御ソフトウェアと同一サイズになると、メモリの節約とはならない。本発明(5)によれば、係る状況の下で、更新(更新用ソフトウェアのサイズ)の柔軟性と、装置の大きさや価格及びダウンロード時間とのトレードオフが可能となる。因みに、更新の柔軟性が低い(バッファサイズが小さい)ほど、装置は小さく安価になり、かつダウンロード時間も短くなる。

【0019】また好ましくは本発明(6)においては、上記本発明(5)において、制御ソフトウェアは各単独で更新可能な複数のモジュールに分割されていると共に、CPUは各モジュールに対するダウンロードと更新の処理を順次繰り返すことにより複数のモジュールを更新する。

【0020】本発明(6)においては、モジュール毎のダウンロードと更新の処理を繰り返す構成により、少ないバッファメモリを使用して大きな制御ソフトウェアの全体を効率良く更新できる。

【0021】また好ましくは本発明(7)においては、上記本発明(5)又は(6)において、更新用ソフトウェアは所定サイズのブロックに分割されていると共に、CPUはブロック単位でダウンロードを要求し、所要サブサイズ分の更新用ソフトウェアを順次バッファメモリに蓄積する。

【0022】本発明(7)においては、CPUはブロック単位でダウンロードを行う構成により、1ブロック分のダウンロードに拘束される時間が短い。またCPUはブロック単位でダウンロードを要求する構成により、ブ

ロック転送の主導権はCPUの側に存在し、このためにCPUは次のブロック転送を要求するまでの間に様々な処理を実行できる。例えば装置の運用に係る通常の通信サービスを実行することが可能であり、この間に、待ち受け状態となって着信を受け付けることも、また発信をすることも可能となる。従って、無線端末装置の本来の機能は損なわれず、その利便性を維持できる。

【0023】また好ましくは、本発明(8)においては、上記本発明(7)において、CPUはブロックデータの転送順を管理すると共に、ブロックデータのダウンロード中に所定のイベントが発生した場合はダウンロード処理を中断する。

【0024】本発明(8)においては、所定のイベントが発生した場合はダウンロード処理を中断する構成により、ダウンロードとダウンロードとの間のみならず、ダウンロード中であってもその処理を開放して必要なイベント処理を速やかに実行できる。所定のイベントとは、例えば発信操作である。またブロックデータの転送順を管理する構成により、ダウンロードがどの時点で中断されても、当該中断されたブロックの始めからダウンロードを再開でき、効率が良い。

【0025】また好ましくは本発明(9)においては、上記本発明(4)において、主メモリの制御ソフトウェアを記憶する部分と、バッファメモリとがフラッシュROMにより構成されている。

【0026】従って、安価なメモリにより、更新用ソフトウェアのダウンロード及びそれによる制御ソフトウェアの更新を安全に行える。

【0027】また本発明(10)のソフトウェア供給装置は、網に接続してデジタル信号を送受信可能な通信インターフェース部と、網に接続する無線端末装置の運用に係る各バージョンの制御ソフトウェア及びそのバージョン情報を格納するメモリと、メモリの記憶情報を更新管理すると共に通信インターフェース部を介して任意の無線端末装置と接続可能な通信制御部とを備え、通信制御部は無線端末装置から通知されたバージョン情報をメモリの最新のバージョン情報と比較して更新の必要性有無を判定すると共に、更新が必要な場合は通知されたバージョンの改訂用に適合する新しい制御ソフトウェアを選択して無線端末装置にダウンロードするものである。

【0028】

【発明の実施の形態】以下、添付図面に従って本発明に好適なる実施の形態を詳細に説明する。なお、全図を通して同一符号は同一又は相当部分を示すものとする。

【0029】図2は実施の形態による移動通信システムの構成を示す図で、図において、100は公衆網(PSTN)、101は移動体交換局(MSC)、102は基地局制御装置(BSC)、103は基地局(BTS)、50はソフトウェア供給装置、60は保守端末、10は移動局装置(無線端末装置、携帯電話機等)である。

【0030】ソフトウェア供給装置50において、51はソフトウェア供給装置50の主制御（呼制御、各種バージョンの制御ソフトウェアの蓄積管理並びにダウンロードの判定及びダウンロード制御等）を行うCPU、52はCPU51が実行する各種アプリケーションプログラムやデータを記憶する主メモリ（MEM）、53は各種バージョンの制御ソフトウェア及びそのバージョン情報等を蓄積格納しているディスク装置（DSK）、54は通信制御部（CIF）、55は保守端末60に接続するシリアルインターフェース部（SIF）、56はCPU51の共通バスである。

【0031】移動局装置10において、11はTDMA方式等による通信制御部、12は送信部、13は送／受分波スイッチ（T/R）、14はアンテナ、15は受信部、16は周波数シンセサイザ（SYN）、17は音声の符号データとPCMデータ間の符号変換を行うコーデック（CDC）、18はPCMデータと音声信号間の変換を行うベースバンド処理部（BBP）、19はマイク（MIC）、20はレシーバ（RCV）、21はブザー（BZ）等の発音体、22は移動局装置10の主制御（呼制御、キー操作制御、制御ソフトウェアのダウンロード／更新制御等）を行うCPU、23はCPU22が実行する各種アプリケーションプログラムやデータを記憶するRAM、マスクROM、フラッシュROM等からなる主メモリ（MEM）、24は液晶表示部やダイヤルキー等を備えるコントロール部（CSL）、25は電源である充電可能なバッテリー部（BT）、26はその充電用端子、27はCPU22の共通バスである。

【0032】ソフトウェア供給装置50は、網側に接続され（又は移動体交換局101等に組み込まれ）、移動局装置10についての各種バージョンの制御（運用）ソフトウェアを蓄積・管理すると共に、移動局装置10からのバージョン比較要求に応じて更新の必要性有／無を判定し、更新が必要な場合は基地局103を介して対応する制御ソフトウェア（更新用ソフトウェア）を移動局装置10にダウンロードする。なお、ソフトウェア供給装置50に対する新たな制御ソフトウェアの供給は保守端末60等から行うことができる。

【0033】移動局装置10に電源投入すると、CPU22は通信制御部11を介して最寄りの基地局103に位置登録し、待ち受け状態となる。この状態で、もし着信があると、CPU22はBPP18に呼出音（リング）のサウンドデータSDを送出し、これによりブザー21が鳴動する。またユーザはこの待ち受け状態から発呼することが可能であり、ユーザがダイヤル発信操作をすると、CPU22は所定の発呼処理に従って基地局103に発信し、通話相手に接続される。なお、この様なCPU22の呼制御（及び後述のソフトウェアダウンロード制御）はCPU22が通信制御部11との間で制御データCDをやり取りすることにより行われる。

【0034】更に、通話中におけるマイク19からの音声信号はBPP18、CDC17、通信制御部11を介して送話データTDに変換（フォーマット）され、更に送信部12、アンテナ14を介して基地局103に送信される。また基地局103からの受信信号は受信部15で受話データRDに変換され、更に通信制御部11、CDC17、BPP18を介して音声信号に変換され、レシーバ20より出力される。

【0035】また、CPU22はこの待ち受け状態を利用して自らソフトウェア供給装置50に発呼び、自己の制御ソフトウェアが最新バージョンか否かを問い合わせることが可能であり、最新バージョンでない場合は、ソフトウェア供給装置50より最新バージョンにステップアップするための制御ソフトウェア（更新用ソフトウェア）がダウンロードされ、こうして自己の制御ソフトウェアを常に最新バージョンに維持できる。以下、詳細に説明する。

【0036】図3～図5は実施の形態による移動局装置のメモリマップを説明する図（1）～（3）で、図3（A）は主メモリ23の一部構成（メモリ空間）を示している。このメモリ空間はマスクROM32と、フラッシュROM（EEPROMでも良い）33と、バッテリバックアップRAM34とから構成され、ここに各メモリの性質に応じたソフトウェアやデータが格納される。

【0037】ところで、一般にソフトウェアの格納アドレスはCPU22の構造やOSの制約により予め決まつており、これに従って各ソフトウェアモジュールの先頭アドレスは固定となっている。携帯電話機を始めとする現状の小型装置にはこの様な制限を持つものが多く、図3（A）もその例に従っている。なお、このCPU22はアドレスを32ビットで表現するものとする。

【0038】更新ソフトウェア41は、図5の更新処理と図9の更新判定処理とからなり、この部分は更新（変更）する必要が無いのでマスクROM32に格納される。更新ソフトウェア41は後述の制御ソフトウェア43を上書きして更新する機能を備え、制御ソフトウェア43が停止している時にのみ動作する。

【0039】バージョン管理領域42は、制御ソフトウェアのバージョン管理情報を書き換え可能で、かつ移動局装置10の電源断続時にも内容を保持する必要があるため、フラッシュROM33上に設けられる。ところで、この様なフラッシュROM33の内容を書き換える場合には、書き込みに先立って消去を行なう必要がある。データの書き込みはバイト単位で行えるが、消去はセクタ単位となる。従って、バージョン管理領域42や後述の制御ソフトウェアの書き換えは、物理的にはセクタ単位で行うことになる。この例のフラッシュROM33では、消去したセクタは全ビット=1になり、書き込みの際は対象ビットを1から0に変更する。

【0040】また、フラッシュROM33では、データ

の書き換えを簡単に行うために、できるだけセクタの先頭となるアドレスにバージョン管理領域42や後述の制御ソフトウェアの各モジュールを配置する。本実施の形態では1セクタが64KBのフラッシュROM33を使用しており、各セクタの先頭アドレスは100000H(Hはヘキサデシマル表示を表す)から100000H(64KB)の間隔となっている。

【0041】移動局装置10の制御(運用)ソフトウェア43は、更新可能かつ不揮発性の要求から同じくフラッシュROM33に格納される。移動局装置10の大きさや価格を抑えるためには、フラッシュROM33のメモリ容量は小さいことが望ましい。そこで、本実施の形態ではフラッシュROM33の格納領域を上書き(セクタ毎の消去、再書き)する方法によって制御ソフトウェア43の記憶領域の再利用を行い、ソフトウェア更新に伴なうフラッシュROM容量の増加を最小限に抑えている。

【0042】一例の制御ソフトウェア43はソフトウェアモジュール(部品)A～Dにより構成されている。モジュールAは例えば通信サービスとは関係無い様な固定的な処理からなり、更新の対象外となっている。モジュールBは主に装置の通信制御処理(呼処理等)に係り、通信サービスの変更／追加／改善等に関係があるため、更新対象となっている。またモジュールC、Dは主にキー操作処理やその他のサービス機能に係り、これも更新対象となっている。なお、この様な制御ソフトウェアを更新する場合には、一般にプログラムサイズが変動するため、各モジュールA～Dの先頭アドレスは、それを見込んだ空き領域を含めて、予め制御ソフトウェアの設計時に決定されている。

【0043】新しい制御ソフトウェアのダウンロードは現行の制御ソフトウェアの実行下で行われる。そのために必要な無線通信機能等の実現には、実際に相互に関連して動作する上記いずれのモジュールA～Dも欠かすことができない。この様な状況の下で、制御ソフトウェア43の更新は、基本的にはモジュール単位で行われる。更にはダウンロードに伴う通信トラヒックの低減を図るために、モジュール内の部分的な更新も可能となっている。なお、制御ソフトウェアをモジュール単位で更新する場合の一般的な留意事項であるが、新しいモジュールBは、モジュールA及び新旧いずれのモジュールC、Dと組み合わせた場合においても、正しく動作するように作成される。

【0044】ダウンロードバッファ44は、ソフトウェア供給装置50よりダウンロードされた新しい制御ソフトウェア(更新用ソフトウェア)を一時的に格納する。ダウンロードバッファ44の内容により現行の制御ソフトウェア43を上書きしている途中で、万一、ダウンロードバッファ44の内容が失われると、現行の制御ソフトウェアは二度と動作できなくなる。また現行の制御ソ

フトウェア43が動作できなくなると、その機能を利用して行われるダウンロードのし直しも不可能となってしまう。そこで、ダウンロードバッファ44にもフラッシュROM33を使用する。

【0045】ダウンロードバッファ44のサイズは、装置のメモリ容量を増大させないために制御ソフトウェア43のサイズよりもできるだけ小さくなるように選択される。但し、本実施の形態では最大のモジュールBがフルに書き換える場合を考慮してモジュールBよりも大きいサイズに選ばれる。その結果、図示の例ではダウンロードバッファ44のサイズは1MB(制御ソフトウェアの1/2)であるが、モジュール数を増やし、各モジュールのサイズを小さくすると、ダウンロードバッファ44のサイズも更に小さくできる。従って、ソフトウェア更新の実現に伴なうフラッシュROM33の容量増加を抑えることが可能であり、移動局装置10の小型低価格化を図ることができる。

【0046】最新のバージョン管理情報がバージョン管理領域42のどのアドレスに記録されているかの情報は、アクセスの利便性から、これをバッテリバックアップRAM34のワークエリア45に記憶しておく。なお、バックアップ電池の残量が不足して記憶が失われた場合は、後述の処理によりバージョン管理領域42から最新のバージョン管理情報を直接に探すことが可能である。

【0047】図3(B)はモジュールBの記憶態様を示している。モジュールBの記憶エリア0.9MBは、モジュールBの本体(プログラム部分)0.8MBと空きエリア0.1MBとからなっている。なお、更新後のモジュールBの本体が記憶エリアの最大サイズ0.9MBを超える様な場合は、後続のモジュールC等を同時に更新して調整可能である。

【0048】図4はバージョン管理領域42の記憶フォーマットを示している。フラッシュROM33上でバージョン管理情報を能率良く管理するために、セクタを2つ設けている。バージョン管理情報は1つ当たり16バイトを使用し、更新の度に新たなバージョン管理情報を後の領域(16バイト)に追記していく。こうして一方のセクタを使い切った時は、他方のセクタの先頭から使い始める。この時、他方のセクタが過去に使われたことがある場合は、当該セクタへのデータ書き込みに先立って、当該セクタを消去する。

【0049】バージョン番号は32ビットの整数で管理し、数値の大きい方が新しいものとする。但し、本実施の形態ではフラッシュROM33を消去すると全ビット=1になることから、これと区別するためにバージョン番号=0FFF FFFF Hは無効とする。32ビットからなるバージョン番号は40億回以上の更新を識別可能であり、フラッシュROM33の書き換え可能回数(10万回程度)を含む移動局装置10の寿命を考えた

場合には、無限とみなせる。

【0050】バージョン番号は、モジュールの組み合わせにより生じる不具合を防ぐため、モジュールではなく、基本的には制御ソフトウェア全体のバージョンを表す。但し、後述の如くある制御ソフトウェアがモジュール毎に段階的に更新される場合があり、この場合は各モジュールの更新毎に制御ソフトウェア全体としてのバージョン番号が更新される。

【0051】上記バージョン番号の他にも、セクタ内の部分的な更新が失敗した様な場合にやり直しができるよう、更新対象である制御ソフトウェア格納領域43内のセクタアドレスと、セクタ内の部分的な更新用ソフトウェアを保存している後述のセクタバッファ44aのアドレスとを記録する。なお、セクタ内の部分的な更新を行わない場合は、セクタアドレスとセクタバッファアドレスとを同じ値にする。セクタ内の部分的な更新を行う場合は、これらが同じ値になることはない。

【0052】更に、「バージョン書込完了フラグ」は上記バージョン管理情報のバージョン管理領域42への書き込みが完了したか否かを表し、また「セクタバッファ書込完了フラグ」は後述セクタバッファ44aへの部分的な更新用ソフトウェアの書き込みが完了したか否かを表し、そして、「更新完了フラグ」は制御ソフトウェアにおける更新対象部分の更新（書き換え）が完了したか否かを表す。なお、これらの詳細は後述の説明により一層明らかとなる。

【0053】図5はダウンロードバッファ44の記憶フォーマットを示している。図において、「モジュール数」は更新するモジュール数を表し、例えば单一のモジュールBを更新する場合は「1」、複数のモジュールC、Dを同時に更新する場合は「2」となる。「モジュール本体」には更新用ソフトウェアのモジュール本体が格納され、少なくとも最大モジュールBに対応する更新用ソフトウェアを格納できる大きさを備える。「アドレス」はモジュール本体の制御ソフトウェア格納領域43における格納先アドレス、「サイズ」はモジュール本体のサイズを表す。「チェックサム」には先頭のモジュール数からチェックサム領域の直前までの各データを1バイトずつ順に加算した結果を格納する。但し、桁あふれは無視する。上記「モジュール本体」以外の各情報はダウンロード時における転送ブロックのフォーマット時に付加されている。転送ブロックのデータフォーマットについては図10に従って後述する。

【0054】なお、ダウンロードバッファ44には単一のモジュールにつき複数の更新用ソフトウェア（各更新単位）をパッキングすることが可能である。この場合の各更新用ソフトウェアは「アドレス1」、「サイズ1」、「更新ソフトウェア本体1」、「アドレス2」、「サイズ2」、「更新ソフトウェア本体2」、…の如く順にパッキングされる。この場合のCPU22は「アド

レス1」、「アドレス2」等の情報に基づき、制御ソフトウェア格納領域43内の飛び飛びの領域に散在するソフトウェアを個々に更新する。これにより広範囲にわたる更新を少ない情報量で能率良く行える。

【0055】「セクタバッファ」44aは、1セクタ(64KB)分のバッファ領域からなり、制御ソフトウェア（モジュール）内のごく一部を更新する場合等において、セクタ内の部分的な更新を安全に行うために使用される。こうして、一例のダウンロードバッファ44のサイズは、モジュール本体(896KB)+管理領域(10B)+セクタバッファ(64KB)からなっており、これをセクタ単位の64KBで切り上げて1MBとしている。

【0056】図6、図7は実施の形態によるダウンロード処理のフローチャート(1)、(2)である。図6において、制御ソフトウェア43の更新が必要か否かの判定は移動局装置10がソフトウェア供給装置50に現用の制御ソフトウェアのバージョン番号を知らせたことを契機としてソフトウェア供給装置50により行われる。この場合に、制御ソフトウェア43の更新は移動局装置10の電源状態が良好で、かつ装置10が使用されていない（待ち受け状態）時に行うのが望ましい。そこで、本実施の形態では移動局装置10が外部の充電装置（不図示）に接続され、かつ所定時間（例えば1時間）以上使用されていなかった時に問い合わせを開始する。なお、移動局装置10が充電装置に接続されていることはバッテリー部25により検出され、CPU22はこれをバス27を介して検知できる。また移動局装置10が所定時間以上使用されていないことは、この区間に着信の受付操作等を含むいかなるキー操作も行われていないことによって検出される。

【0057】上記条件を満足すると、図示しないが、CPU22は、制御ソフトウェア43に従いままでソフトウェア供給装置50に自動的に発呼する。この発呼は網側が提供する通信サービスに対する特番発呼でも良いし、又はソフトウェア供給装置50を着信先とする様な通常の発呼でも良い。そして、呼が接続されるとステップS41の処理に入力する。ステップS41ではCPU22は「処理種別=1：バージョン比較要求」、「バージョン番号」をソフトウェア供給装置50に送信する。「バージョン番号」は自己の現用の制御ソフトウェア43のバージョン番号である。

【0058】ソフトウェア供給装置50は、バージョン比較要求を受け付けたことにより、ステップJ1では受信したバージョン番号が当該移動局装置10について移動通信システムが管理するところの最新バージョンか否かを判別する。最新でない場合は、ステップJ2で最新バージョン又は最新バージョンにステップアップするための適切な更新用ソフトウェア（モジュール）を選択する。ステップJ3では当該更新用ソフトウェアの転送ブ

ロック数を取得する。また上記ステップJ 1の判別で最新バージョンの場合はステップJ 4で転送ブロック数=0を取得する。ステップJ 5では転送ブロック数=N(但し、N≠0の場合は新たな制御ソフトウェアのバージョン番号を含む)を移動局装置10に返信する。

【0059】CPU22は、転送ブロック数を受信したことにより、ステップS 4 2では転送ブロック数>0か否かを判別する。転送ブロック数=0の場合は、移動局装置10が最新バージョンで運用中であることを表すため、ソフトウェア供給装置50との間の呼を切断してステップS 4 3に進み、移動局装置10としての通常の運用処理を実行する。また転送ブロック数>0の場合は、図7のステップS 4 5に進み、以下のダウンロード処理を開始する。

【0060】図7において、ダウンロード処理は、移動局装置10が転送ブロックを要求し、ソフトウェア供給装置50がその応答として転送ブロックを送信するという手順で行われる。即ち、ステップS 4 5ではCPU22は「処理種別=2：転送要求」、「バージョン番号」、「転送ブロック番号」を送信する。この転送要求には、特定の転送ブロックを簡単かつ一意に特定できるように、転送要求に係る制御ソフトウェアのバージョン番号と転送ブロック番号とが含まれる。転送ブロックには1ずつ増加する一連の番号が付与されており、これを転送ブロック番号と呼ぶ。各バージョン(モジュール)における先頭の転送ブロックの転送ブロック番号を1とする。

【0061】ソフトウェア供給装置50は、転送要求を受け付けたことにより、ステップJ 7で対応する転送ブロックを移動局装置10に返送する。その際には、ソフトウェア供給装置50は転送要求に含まれるバージョン番号及び転送ブロック番号の情報をキーとして、管理テーブルから対応する転送ブロックを読み出す等の単純な処理により、速やかに転送ブロックを選択できる。

【0062】図10に実施の形態による転送ブロックのイメージ図を示す。例えばモジュールBを更新する場合は、ソフトウェア供給装置50は予め更新対象の必要な1又は2以上の更新用ソフトウェアを収集して「モジュール情報」の欄に展開する。この展開は上記ダウンロードバッファ44で述べた各フォーマット情報に対応して、更新用ソフトウェア毎に「アドレス」、「サイズ」、「更新用ソフトウェア本体」の各情報をパッキングする。更に、その先頭には「モジュール数=1」の情報を付加し、またその最後尾には「モジュール数」と「モジュール情報」とについて求めた「チェックサム」の情報を付加する。そして、これらを各所要サイズの転送ブロックにブロック分割し、これらを転送ブロック番号との対応にメモリ(DSK53等)に格納しておく。転送ブロックのサイズは、無線通信速度が9600bpsの場合は、1ブロックの転送時間が30秒以下となる

様に32KB以下に選択される。

【0063】なお、ダウンロードバッファ44に対して複数のモジュールC、D等を一举に転送することが可能であり、この場合は「モジュール数=2等」とし、かつ「モジュール情報」の欄にはモジュールC、Dについての各更新用ソフトウェア毎に「アドレス」、「サイズ」、「更新用ソフトウェア本体」の各情報をパッキングする。

【0064】図7に戻り、CPU22は、転送ブロックを受信したことにより、ステップS 4 6では該転送ブロックをダウンロードバッファ44に蓄積する。ステップS 4 7では転送ブロック番号に+1する。ステップS 4 8では転送ブロック番号>Nか否かを判別し、NOの場合はステップS 4 9でタイマ(例えば3分)をスタートさせる。ステップS 5 0では該タイマによる割り込みを許可とし、ステップS 5 1では現用の制御ソフトウェア43内の通常の運用処理に制御を移す。その際には、ソフトウェア供給装置50との間の呼は一旦切断(又は保留に)され、移動局装置10は待ち受け状態に戻る。この間に、移動局装置10は通常の発/着信を行う事が可能であり、もし発/着信が行われた場合には、通話終了までタイマの進行は停止される。

【0065】やがて、待ち受け状態のままで3分を経過すると、上記タイマがタイムアウトし、そのタイマ割込処理により制御はステップS 5 2に移る。ステップS 5 2ではタイマ割込を不許可にし、かつ前記呼を切断(又は保留)していたソフトウェア供給装置50に呼を接続して上記ステップS 4 5に戻る。以下同様にして進み、やがて、ステップS 4 8の判別で転送ブロック番号>Nになると、ダウンロードバッファ44へのダウンロード終了である。処理はステップS 5 3に進み、ソフトウェア供給装置50との間の呼を切断すると共に運用中の制御ソフトウェア43を停止し、図8の更新処理(更新ソフトウェア41)に制御を移す。

【0066】図8は実施の形態による更新処理のフローチャートである。ステップS 2 1では新たなバージョン管理情報の書込先領域(16バイト分)の内容がOF…FHか否かを判別する。なお、最新のバージョン管理情報の所在に関する情報はバッテリバックアップRAM34のワークエリア45に記憶されており、通常はここに記録が残っているので、その次の書込先領域を選択すれば、その内容はOF…FHとなり、処理はステップS 2 5に進む。但し、バックアップ電池の残量が不足して記録が失われた様な場合、又は既に使用されたセクタを新たに使用する様な場合には、この判別はNOとなる場合がある。NOの場合はこの書込先領域が既に使用されているため、更にステップS 2 2においてこの書込先領域はセクタの先頭か否かを判別する。先頭でない場合はステップS 2 4で次の書込先領域を選択してステップS 2 1に戻る。また先頭の場合はステップS 2 3でこの書込

先領域を含むセクタを消去し、消去後にステップS 2 1に戻る。

【0067】上記ステップS 2 1の判別で新たな書き込み領域が検出されると、ステップS 2 5では「バージョン番号」の欄にダウンロードされた制御ソフトウェアのバージョン番号を書き込む。ステップS 2 6では「セクタアドレス」の欄に更新対象となる制御ソフトウェア記憶領域4 3のセクタアドレスを書き込む。また、もしセクタ内の部分的な更新をする場合には、「セクタバッファアドレス」の欄に部分的な更新ソフトウェアを格納しているセクタバッファ4 4 aのアドレスを書き込む。なお、部分的な更新をしない場合は「セクタバッファアドレス」の欄に「セクタアドレス」の欄と同じ内容を書き込む。ステップS 2 7ではバージョン書込完了フラグに0(完了)を書き込む。

【0068】ステップS 2 8ではセクタ内の部分的な更新を行うか否かを判別する。部分的な更新を行わない場合は、ステップS 3 2で更新対象の連続する又は飛び飛びの複数のセクタをダウンロードバッファ4 4に格納されている各モジュール本体の内容で書き換える。書き換えは、更新対象領域の各セクタを消去し、その上に更新用ソフトウェアを書き込む。そして、ステップS 3 3に進む。

【0069】また上記ステップS 2 8の判別でセクタ内の部分的な更新を行う場合は、ステップS 2 9でセクタバッファ4 4 aに更新内容(1セクタ分)を取得する。具体的に言うと、更新対象のセクタから更新前の制御ソフトウェアデータを読み出すと共に、その一部分をダウンロードバッファ4 4に格納された部分的な更新データによって書き換え、これをセクタバッファ4 4 aに書き込む。好ましくは、この様なデータ書き換え処理の途中で発生する様な装置の電源断にも有効に対処するため、データの転送処理はバッテリバックアップされたワークエリア4 5を介して行われる。ステップS 3 0ではセクタバッファ書込完了フラグに0(完了)を書き込む。ステップS 3 1では更新対象の1セクタ分の内容をセクタバッファ4 4 aの内容で書き換える。書き換えは、更新対象の1セクタを消去し、その上にセクタバッファ4 4 aの内容を書き込む。ステップS 3 3では更新完了フラグに0(完了)を書き込む。そして、ステップS 3 4では更新された制御ソフトウェア4 3を起動する。

【0070】この様に更新処理の各段階が完了する毎に対応する完了フラグの記録を取るのは、更新ソフトウェア(モジュール等)が制御ソフトウェアの格納領域4 3とダウンロードバッファ4 4のいずれに存在していたかを後になって各段階毎に正確に把握できるようにするためにある。

【0071】なお、図示しないが、これらの一連の更新処理の途中で、万一、フラッシュROM3 3への書き込みや消去動作においてエラー(ハードウェア的に検出可

能)が発生した場合には、コンソール部2 4に移動局装置1 0の修理が必要である旨を表示して処理を停止する。

【0072】また、上記ダウンロードが完了した時点と更新が完了した時点の双方にて、データ誤り(チェックサム)の検出を行い、問題があれば再試行することにより、更新時の事故を最低限に抑えることができる。

【0073】以上はダウンロードバッファ4 4に格納可能な1回分のダウンロード処理及びその更新処理を述べたが、ダウンロードバッファ4 4のサイズに制限があるため、1回の更新では必ずしも制御ソフトウェア4 3の全体が最新バージョンになるとは限らない。即ち、あるバージョンの移動局装置1 0が最新バージョンとなるためには数回にわたるモジュールの更新を要する場合がある。この場合は、移動局装置1 0は上記のダウンロード及び更新の処理を繰り返すことでダウンロードバッファ4 4のサイズを越える様な制御ソフトウェア全体の更新が可能となる。

【0074】例えば移動局装置1 0は、ある制御ソフトウェアの更新完了後に再びステップS 4 1のバージョン比較要求処理へ移行することにより、上記複数回の更新が必要であったり、又はある制御ソフトウェアのダウンロード中に該制御ソフトウェアの最新版が更新されていた様な場合であっても、これを速やかに再更新することが可能である。この場合の制御ソフトウェア4 3は、更新を行った直後の場合には、まずバージョン比較要求処理を行うように作り込まれる。更新直後であるか否かの状態はバッテリバックアップRAM3 4に記録しておく。もしその内容が失われた場合は、バージョン比較要求を行わずに通常の運用処理を開始する。この場合は更新が遅れることになるが、移動局装置1 0の動作に支障はない。

【0075】图9は実施の形態による更新判定処理のフローチャートである。本実施の形態では、上記制御ソフトウェア4 3の更新処理中は移動局装置1 0に対する一切の操作を受け付けないようにするが、電池残量の不足や電池の取り外しによって更新処理が中断してしまうことは避けられない。このため移動局装置1 0が起動(電源投入等)した時には、制御ソフトウェア4 3を起動する前に、それが正しく動作可能であるか否か、即ち、更新処理の途中でなかったか否かを判定するために、まずはこの更新判定処理を行う。

【0076】ステップS 1 1では最新のバージョン管理情報が既知か否かを判別する。最新のバージョン管理情報の所在については、バッテリバックアップRAM3 4のワークエリア4 5に記憶されており、通常は記録が残っているので、この判別は既知となる。しかし、バックアップ電池の残量が不足して記録が失われた様な場合にはステップS 1 2で最新のバージョン管理情報を探す。

【0077】最新のバージョン管理情報を見つけるに

は、まずバージョン管理情報を格納している2つのセクタの先頭領域（バージョン番号欄）を調べる。いずれかのセクタが0FFF FFFF FH（消去状態）であれば、そうでない方のセクタに最新のバージョン管理情報が存在する。また双方共に0FFF FFFF FHでない時は、バージョン番号の値の大きい方に存在する。こうして、一方のセクタを特定した後は、先頭から順にバージョン番号を調べていくことにより最新のバージョン管理情報を見つけることができる。この時、セクタの末尾に到達すれば、それが最新のバージョン管理情報であるし、また途中で0FFF FFFF FHが見つかれば、その手前に最新のバージョン管理情報が存在する。

【0078】ステップS13では最新のバージョン管理情報の更新完了フラグ=0（完了）か否かを判別する。更新完了の場合は図8のステップS34に進み、制御ソフトウェア43を起動する。また更新完了でない場合は更にステップS14でバージョン書込完了フラグ=0（完了）か否かを判別する。バージョン書込完了でない場合は図8の更新処理（ステップS21）に制御を移す。なお、上記最新のバージョン管理領域は既知であるから、上記ステップS21に制御を移す代わりに、ステップS25に制御を移しても良い。またバージョン書込完了の場合は更にステップS15でセクタ内の部分的な更新か否かを判別する。既にバージョン書込完了となっているので、この判別は「セクタアドレス」欄と「セクタバッファアドレス」欄の情報を調べることで行える。部分的な更新でない場合は図8のステップS32に進み、更新対象の複数のセクタを書き換える。また部分的な更新の場合は、更にステップS16でセクタバッファ書込完了フラグ=0（完了）か否かを判別する。セクタバッファ書込完了でない場合は図8のステップS29に進み、セクタバッファ44aに更新内容を取得する。またセクタバッファ書込完了の場合は図8のステップS31に進み、更新対象の1セクタを書き換える。

【0079】図1.1～図1.3は実施の形態によるダウンロード／更新処理のイメージ図（1）～（3）で、図1.1はモジュールB, C, Dが更新対象となった場合のタイミングチャートを示している。ダウンロードバッファ44の大きさによる制限があるため、全モジュールを同時に更新することはできない。ここではモジュールBを更新した後にモジュールC, Dを更新する。更新を2段階に分けて行うため、制御ソフトウェア43のバージョン番号も2回更新されている。

【0080】ダウンロードに要する時間は更新対象のサイズに比例する。特に更新対象が大きい場合には、長時間にわたって移動局装置10が使用できなくなることは避けなければならない。この移動局装置10は通信経路（チャネル）を1系統しか持たないため、通信中は着信を受け付けることができなくなる。通信中の本装置10に対して発信すると、話中となってしまう。ダウ

ロード時も例外ではない。着信を全く妨げないようにすることは不可能であるが、長時間連続して着信不可能となるないようにすることは可能である。ここでは、転送ブロックの転送毎に通常運用を行う時間（例えば3分以上）を設けることにより、着信が入り込む隙間を空けておく。この間に着信があった場合は、続くダウンロードを中断（延期）する。なお、ユーザはこの間に発信することも可能である。

【0081】一例の時間配分を具体的に述べると、この移動局装置10の無線通信速度は例えば9600 bpsである。この場合の転送ブロックのサイズは例えば32KB以下として、30秒以内で転送できるようにする。1ブロック転送完了後の休止時間は例えば3分以上とする。もし、このようなブロック分割と休止を行わない場合は、1MBのダウンロードでは約15分間にわたって着信が不可能となる。これを30秒の転送と3分の休止の繰り返しとすると、ダウンロード完了までには約112分かかるてしまうが、一般にこの様なソフトウェア更新の遅れは問題とはならない。むしろ、多くの場合に、加入者は最初に話中であった時には1～3分程度待って発信し直すことを考慮すると、3分以上の休止時間を設けることで装置運用上の悪影響は少ないと言える。

【0082】また、ユーザの発／着信操作を妨げないようにするために、ダウンロード中（ブロック転送中）であってもキー入力等を受け付けるようにして、キー入力操作された時には速やかにダウンロードを中断することが好ましい。

【0083】また、ダウンロードによる通信中は、待ち受け状態であっても大幅に消費電力が上昇する。ダウンロードによって装置10の公称の待ち受け時間が短くなることは好ましくないため、ダウンロードによって電池を消費しないように、好ましくは移動局装置10が充電装置から外された時にはダウンロードを中断する。

【0084】上記様々な理由でダウンロードが中断された場合には、その続きから再開できるようにすると無駄が多い。これを簡単に実現するため、ダウンロード対象を小さな転送ブロックに分割して転送している。中断されたダウンロードの再開は転送ブロックの単位で行う。再開する転送ブロックを決めるための進行状況はバッテリバックアップRAM34において記録・管理する。その内容が失われた場合には、始めからやり直す。なお、ダウンロードの中止と再開によって転送ブロックの抜けが発生した様な場合は、ダウンロードバッファ44におけるチェックサムの検査により誤りを検出できる。

【0085】図1.2は1段階目のモジュールBの更新処理のイメージを示している。なお、図では稼働中のソフトウェアに網点を付記してある。旧モジュールBを含む現行の制御ソフトウェア43によって新しいモジュールBをダウンロードした後、現行の制御ソフトウェア43

を一旦停止し、この間に更新ソフトウェア41がモジュールBを更新する。モジュールBの更新が完了した時点での制御ソフトウェア43のバージョン番号を更新し、制御をモジュールBが更新された新しい制御ソフトウェア43に戻す。

【0086】図13は2段階目のモジュールC、Dの更新処理のイメージを示している。モジュールC、Dはダウンロードバッファ44に収まる大きさであるため、同時に更新できる。旧モジュールC、Dを含む現行の制御ソフトウェア43によって新しいモジュールC、Dをダウンロードした後、現行の制御ソフトウェア43を一旦停止し、この間に更新ソフトウェア41がモジュールC、Dを更新する。モジュールC、Dの更新が完了した時点での制御ソフトウェア43のバージョン番号を更新し、制御をモジュールC、Dが更新された新しい制御ソフトウェア43に戻す。なお、上記複数のモジュールC、Dを同時に更新できる機能を利用して、該複数のモジュールC、Dにまたがる様な大規模なソフトウェア変更を制御ソフトウェア43に加えることも可能となる。

【0087】本実施の形態によれば、ソフトウェア供給装置50は各移動局装置10からのブロック転送要求に応じて対応する転送ブロックを送信するだけでなく、各移動局装置10における更新の進行状況等は各々の移動局装置10が自分自身で管理する。従って、ソフトウェア供給装置50の処理負荷を大幅に抑えることができる。また移動局装置10からソフトウェア供給装置50に対して要求を出し、それに対する応答によって更新対象を受け取る方法は、網側に一斉通知の仕組みを必要としないため、通信システム全体に関わるような変更を加える必要はなく、移動局装置10及び対応するソフトウェア供給装置50をシステムに追加するだけで、スムーズな導入が期待できる。

【0088】なお、上記実施の形態ではTDMA方式による移動通信システムへの適用例を述べたが、本発明は他の様々な通信方式(CDMA等)の移動通信システムにも適用できる。また、この様な大規模な移動通信システムのみならず、本部局と複数の無線端末局とが無線接続可能な他の一般の無線システム(業務用無線通信システム等)にも適用できる。

【0089】また、上記本発明に好適なる実施の形態を述べたが、本発明思想を逸脱しない範囲内で各部の構成、制御、及びこれらの組合せの様々な変更が行えることは言うまでも無い。

【0090】

【発明の効果】以上述べた如く本発明によれば、既存の移動通信システムに殆ど変更を加えることなく、ソフトウェア供給装置と、それに対応した移動局装置を導入することで、制御ソフトウェアの自動更新が容易に可能となる。また少ないメモリで大きな制御ソフトウェアの更新を効率良く安全に行え、装置の小型化、低価格化を容

易にする。

【図面の簡単な説明】

【図1】本発明の原理を説明する図である。

【図2】実施の形態による移動通信システムの構成を示す図である。

【図3】実施の形態による移動局装置のメモリマップを説明する図(1)である。

【図4】実施の形態による移動局装置のメモリマップを説明する図(2)である。

【図5】実施の形態による移動局装置のメモリマップを説明する図(3)である。

【図6】実施の形態によるダウンロード処理のフローチャート(1)である。

【図7】実施の形態によるダウンロード処理のフローチャート(2)である。

【図8】実施の形態による更新処理のフローチャートである。

【図9】実施の形態による更新判定処理のフローチャートである。

【図10】実施の形態による転送ブロックのイメージ図である。

【図11】実施の形態によるダウンロード/更新処理のイメージ図(1)である。

【図12】実施の形態によるダウンロード/更新処理のイメージ図(2)である。

【図13】実施の形態によるダウンロード/更新処理のイメージ図(3)である。

【図14】従来技術を説明する図である。

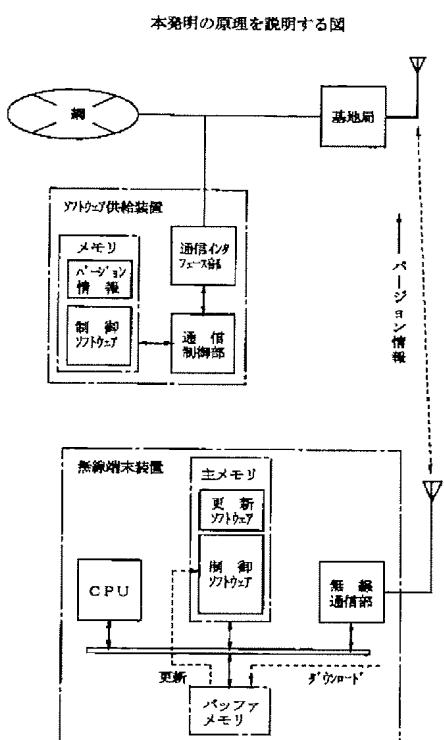
【符号の説明】

- 1 0 移動局装置
- 1 1 通信制御部
- 1 2 送信部
- 1 3 送/受分波スイッチ(T/R)
- 1 4 アンテナ
- 1 5 受信部
- 1 6 周波数シンセサイザ(SYN)
- 1 7 コーデック(CDC)
- 1 8 ベースバンド処理部(BBP)
- 1 9 マイク(MIC)
- 2 0 レシーバ(RCV)
- 2 1 発音体
- 2 2 CPU
- 2 3 主メモリ(MEM)
- 2 4 コンソール部(CSL)
- 2 5 バッテリ一部
- 2 6 充電用端子
- 2 7 共通バス
- 3 2 マスクROM
- 3 3 フラッシュROM
- 3 4 バッテリバックアップRAM

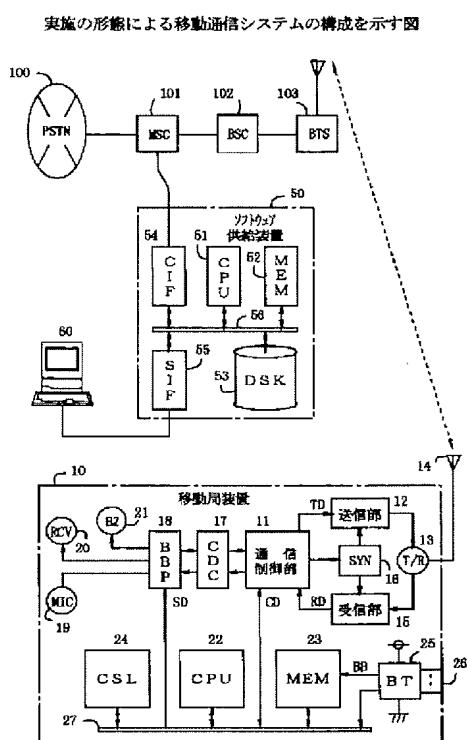
4 1 更新ソフトウェア
 4 2 バージョン管理領域
 4 3 制御ソフトウェア
 4 4 ダウンロードバッファ
 4 5 ワークエリア
 5 0 ソフトウェア供給装置
 5 1 CPU
 5 2 主メモリ (MEM)
 5 3 ディスク装置 (DSK)

5 4 通信制御部 (CIF)
 5 5 シリアルインターフェース部 (SIF)
 5 6 共通バス
 6 0 保守端末
 100 公衆網 (PSTN)
 101 移動体交換局 (MSC)
 102 基地局制御装置 (BSC)
 103 基地局 (BTS)

【図1】

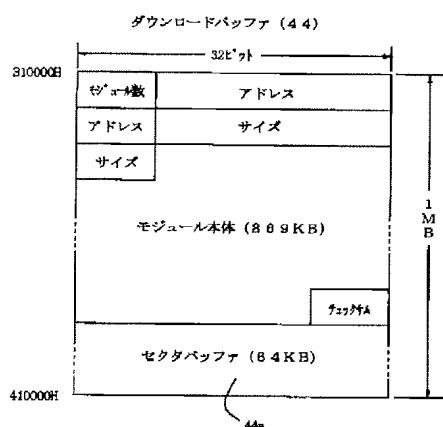


【図2】

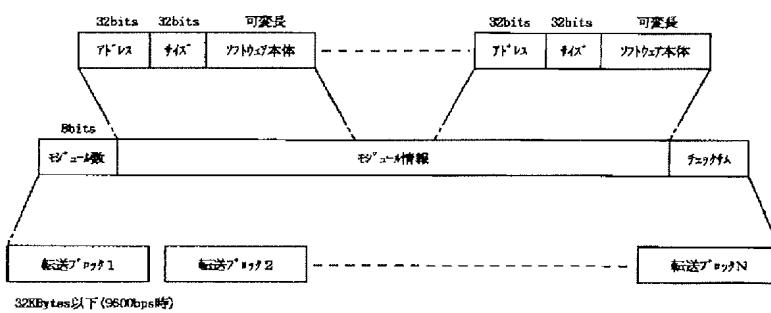


【図5】

実施の形態による移動局装置のメモリマップを説明する図(3)

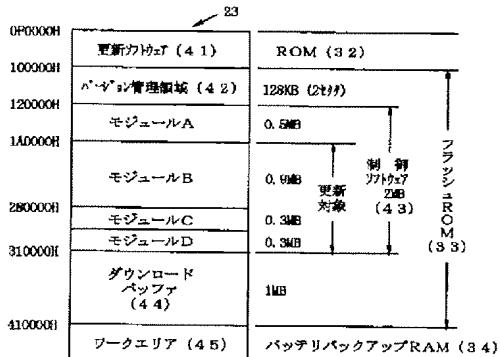


実施の形態による転送ブロックのイメージ図

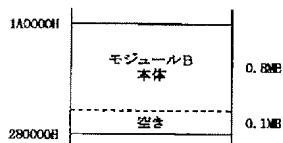


【図3】

実施の形態による移動局装置のメモリマップを説明する図(1)



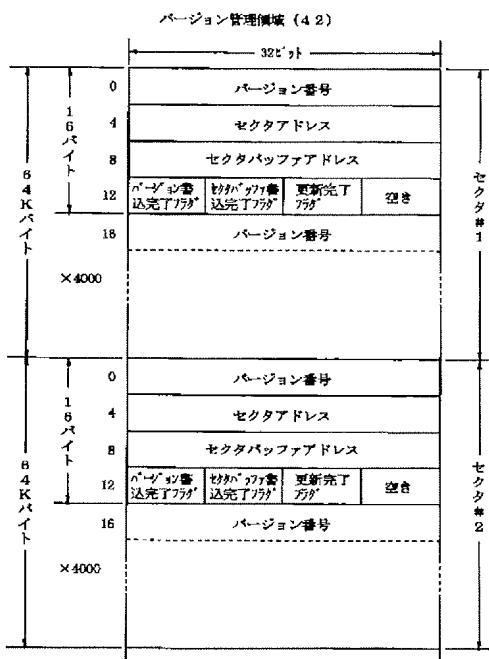
(A) メモリ空間



(B) モジュールB

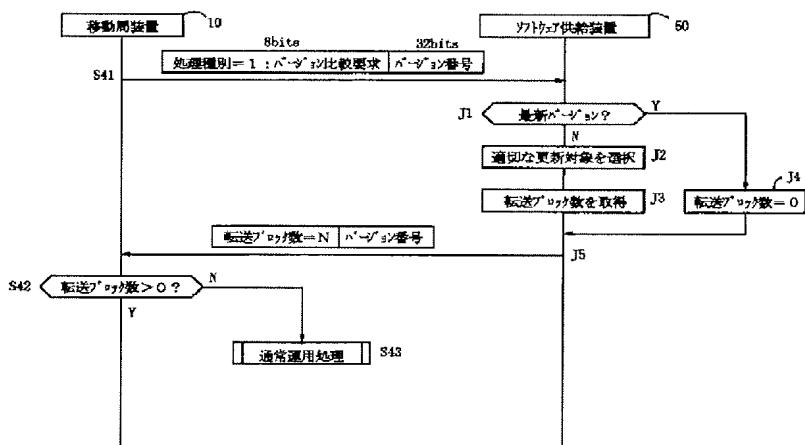
【図4】

実施の形態による移動局装置のメモリマップを説明する図(2)



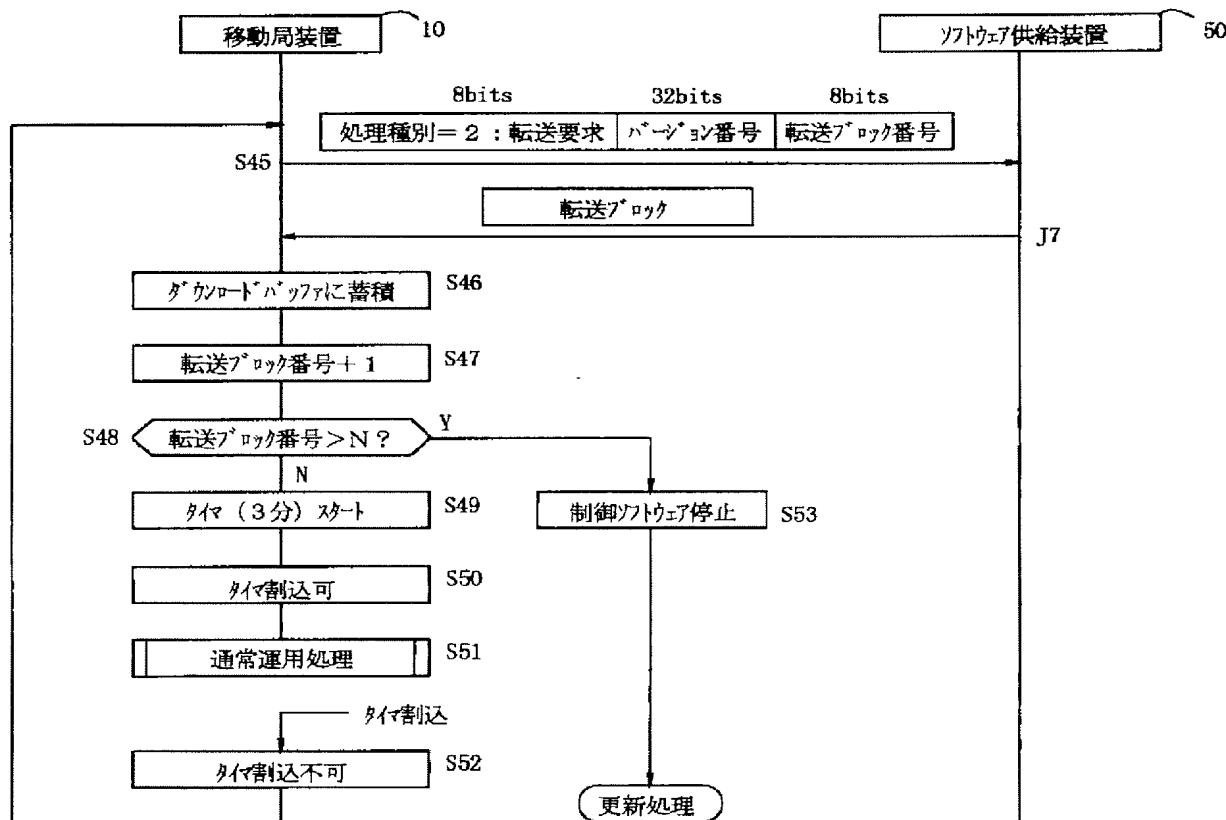
【図6】

実施の形態によるダウンロード処理のフローチャート(1)



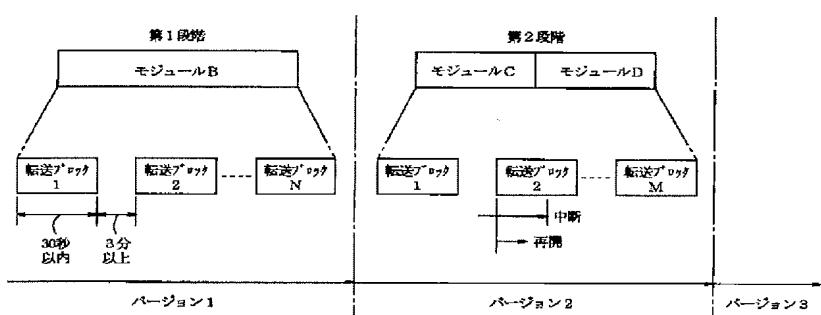
【図7】

実施の形態によるダウンロード処理のフローチャート(2)



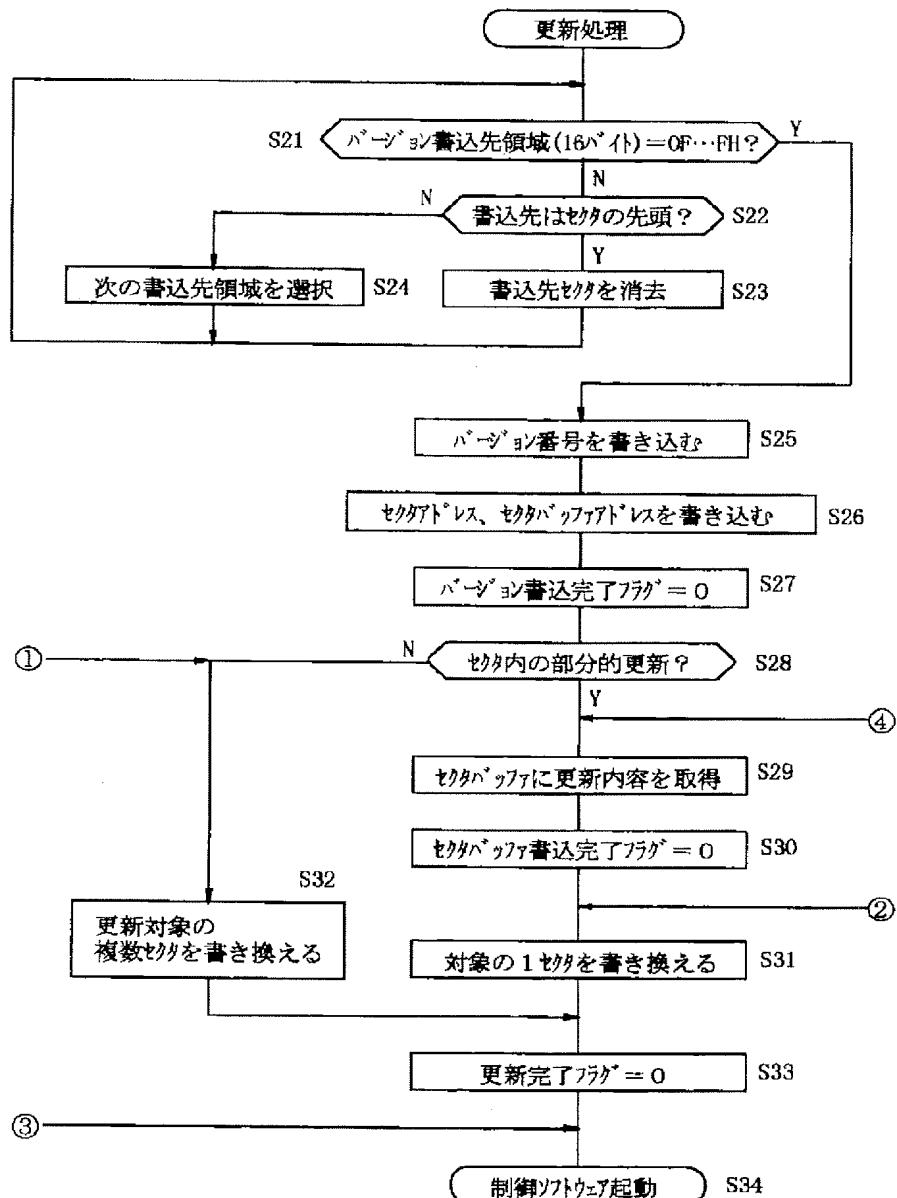
【図11】

実施の形態によるダウンロード／更新処理のイメージ図(1)



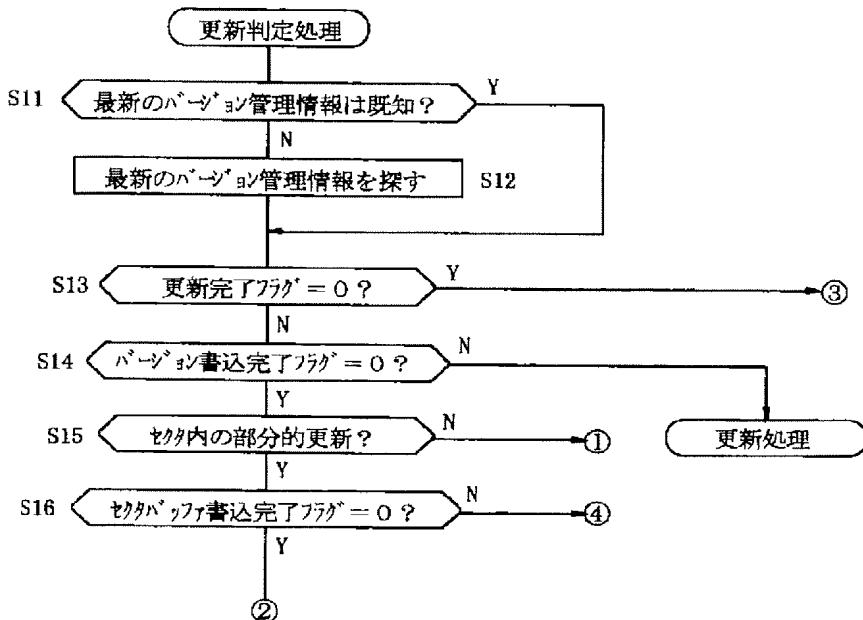
【図8】

実施の形態による更新処理のフローチャート

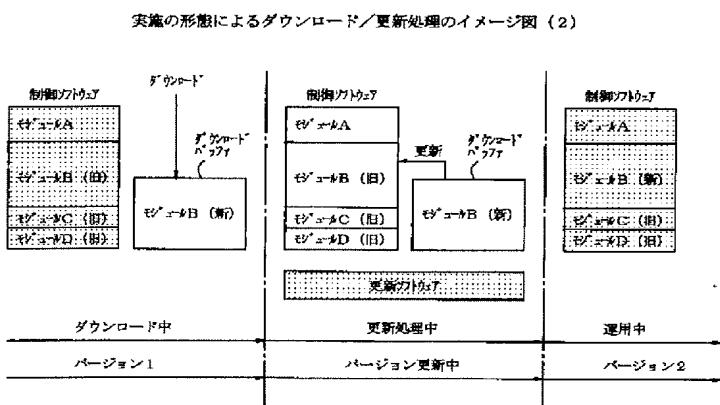


【図9】

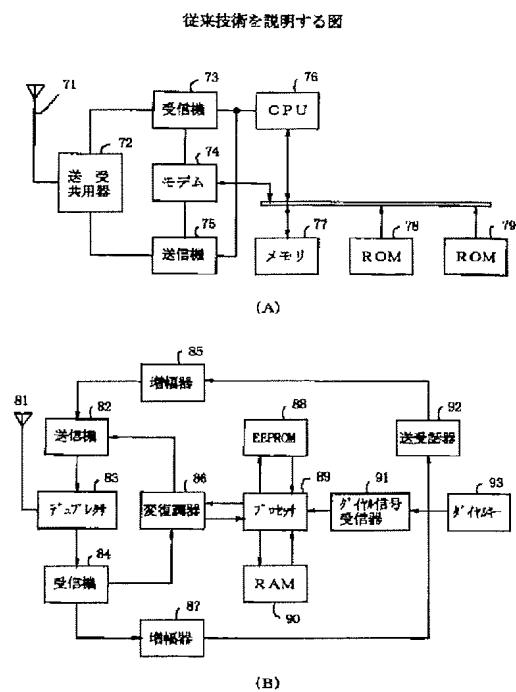
実施の形態による更新判定処理のフローチャート



【図1-2】



【図1-4】



【図1.3】

実施の形態によるダウンロード/更新処理のイメージ図（3）

